

C. REMARKS

Applicants respectfully request reconsideration of the outstanding rejections and reexamination of the present application in light of the following amendments and remarks.

Status of the Claims

Claims 1-16 are currently pending in the application. Claims 3, 9, and 13 are currently amended. Claims 17-19 are canceled.

35 USC 101

The Office Action rejects claims 13-16 under 35 USC 101 as directed to non-statutory subject matter. [Office Action, p. 2] In particular, the Office Action states:

As provided on page 9 of the specification, a computer readable medium includes transmission media. Claims drawn to the components involving signals encoded with functional descriptive material do not fall within any of the categories of statutory subject matter as set forth in 35 USC 101, and are therefore, ineligible for protection. [Office Action, p. 2]

Applicants respectfully note that paragraph 0021 of the specification of the present application distinguishes volatile and non-volatile computer readable media from transmission media. The examples of volatile and non-volatile computer readable media, such as a floppy disk, a flexible disk, a hard disk, magnetic tape, a compact disc ROM or any other optical medium, other types of ROMs, a mass storage device, and dynamic memory, are directed to statutory subject matter. Applicants amend the preamble of claim 13 to limit the computer readable medium to a volatile or non-volatile computer readable medium, which removes transmission media, involving signals encoded with functional descriptive material, from claims 13-16. In view of the amendment to claim 13 to clarify that the computer readable medium of claims 13-16 does not include transmission media, the rejection of claims 13-16 under 35 USC 101 is overcome and Applicants respectfully request withdrawal of the rejection and allowance of the claims.

35 USC 112

The Office Action rejects claim 17 under 35 USC 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. [Office Action, p. 2] Claim 17 is canceled.

Alleged Anticipation under 35 USC 102(e)

The Office Action rejects claims 1, 3-4, 6-7, 9-10, 12, and 17-19 under 35 USC 102(e) as being anticipated by Beliveau et al. (US Patent 6,731,598)(herein referred to as Beliveau). [Office Action, p. 3] Claims 17-19 are canceled. Applicants respectfully traverse the rejections of pending claims 1, 3-4, 6-7, 9-10, and 12 under 35 USC 102(e). “A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. *Verdegaal Bros. v. Union Oil Co. of California*, 2 USPQ2d 1051, 1053 (Fed Cir. 1987). Furthermore the reference must be an enabling disclosure of each and every element as set forth in the claim. *In re Hoecksma*, 158 USPQ 596, 600 (CCPA 1968); *In re LeGrive*, 133 USPQ 365, 372 (CCPA 1962). Because Beliveau does not teach each and every element of claims 1, 3-4, 6-7, 9-10, or 12 or enable each and every element of these claims, these claims are not anticipated, the rejection should be withdrawn, and the claims should be allowed.

Claims 1, 7

The Office Action rejects claims 1 and 7 on the same grounds. Claim 1 reads:

Claim 1 (Original): A method for redirecting connection requests at an operating system kernel level comprising:

receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service;

setting up all of said sockets in said list of sockets to reference each other in said operating system kernel; and

responsive to receiving an incoming connection request for a first socket from said list of sockets that is full, redirecting said connection request to a second socket in said list of sockets that is not full, such that said operating system kernel redirects said connection request to said second socket providing said same service as said first socket.

First, Applicants respectfully assert that the Beliveau does not anticipate claims 1 and 7 because Beliveau does not teach or enable each and every element of receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of a sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service.

The Office Action cites column 7, line 62-column 8, line 17 of Beliveau as describing “a socket list is maintained which contains server socket information including service related information, e.g. there was the list of all FTP servers in the cluster of servers each having its own socket in the list” and cites this portion of Beliveau as reading on the claimed element of receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service. [Office Action, pp. 3-4]

Applicants respectfully assert that regardless of whether Beliveau’s description supports the Examiner’s assertion that describes “a socket list is maintained which contains server socket information including service related information”, maintaining a socket list that contains server socket information does not teach or enable a socket option call with a list of sockets or receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service.

In addition, Applicants respectfully assert that in view of the actual teachings in Beliveau, Beliveau does not teach or enable each and every element of receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets

in said list of sockets will provide said same service. In considering Beliveau as whole, Applicants note that column 7, line 64-column 8, line 3 of Beliveau specify:

when a server uses an API to open a server socket (for example, 80) for a VIP address at step 52, the system call determines that it is a socket for a VIP address. At step 53, the Framework then requests one of the Fragmenter/De-fragmenters to update the list of IPC ports with this new server socket. For the same combination of VIP address/server socket, there may be many IPC ports.

Column 8, lines 23-25 of Beliveau describe the situation where a server uses an API to open a server socket, in that “when an application server such as HTTP-2 needs to establish a socket between itself and a remote client, it first opens a client socket at step 61.” Column 8, lines 4-15 of Beliveau describe:

Therefore, at 54, a server socket list is distributed and shared between all Fragmenter/De-fragmenters. When a packet comes from any source IP address and reaches a Fragmenter/De-fragmenter, the process extracts the destination VIP address and destination socket (for example, 80) from the packet. The Fragmenter/De-fragmenter then finds, through the server socket list, a valid application server. If a plurality of servers can serve this VIP address and server socket combination, the Fragmenter/De-fragmenter selects one of them. For example, if there are six different processors with FTP servers running on them for this VIP address, then the Fragmenter/De-fragmenter selects one of them.”

Applicants respectfully submit that Beliveau’s description of a server using an API to open a server socket for a VIP address, does not however, teach or enable a socket option call with a list of sockets. In addition, Beliveau’s description of a socket call through an API triggering a Framework to request a Fragmenter process to add a newly opened socket for a VIP address to an existing list of IPC ports and VIP address/server socket combinations does not teach a socket option call with a list of sockets. Therefore, Beliveau does not teach or enable the claimed a socket option call with a list of sockets.

In addition, clearly, when Beliveau is considered as a whole, the list of VIP address/server sockets combinations is not compiled responsive to a single socket option call. Applicants respectfully submit that merely because the compiled list of VIP

address/server socket combinations in Beliveau may include multiple servers that can serve a same VIP address and server socket combination, does not teach an operating system kernel receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing the operating system kernel that all of the sockets in the list of sockets will provide the same service.

Therefore, because Beliveau does not teach each and every element of receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing the operating system kernel that all of the sockets in the list of sockets will provide the same service, Beliveau does not anticipate claims 1 and 7 and the claims should be allowed.

Second, Applicants respectfully assert that Beliveau does not anticipate claims 1 or 7 because Beliveau does not teach and enable each and every element of responsive to receiving an incoming connection request for a first socket from said list of sockets that is full, redirecting said connection request to a second socket in said list of sockets that is not full.

The Office Action cites column 8, lines 13-17 as describing "if a socket is full, inherently, since the system assigns services based upon processor load or delays, the system will assign the service to a second socket" and as reading on the element of responsive to receiving an incoming connection request for a first socket from said list of sockets that is full, redirecting said connection request to a second socket in said list of sockets that is not full. [Office Action, p. 4]

Applicants respectfully assert that regardless of the Examiner's assertion as to what is inherent in Beliveau, Beliveau fails to teach responsive to receiving an incoming connection request for a first socket from the list of sockets. Beliveau describes receiving a packet at a Fragmenter specified to receive packets for the source IP address within the packet, where the Fragmenter extracts a VIP address and socket, finds one or more valid applications servers for the VIP address in the socket server list, and if there is more than one valid application server, selects one of the servers.

Beliveau, col. 7, lines 10-25, col. 8, lines 5-18. As noted in the Office Action, Beliveau

describes that in selecting one of the servers, “the selection may be based on a round-robin selection or may be enhanced to consider processor load, delays, or other factors.” Beliveau clearly describes that each incoming request is for one of one or more virtual IPs supported within the Framework, and not for any particular socket. *Beliveau*, col. 1, lines 38-39, col. 4, line 66-col. 5, line 7, col. 7, lines 10-25, col. 8, lines 5-18. Therefore, Beliveau does not teach or enable receiving an incoming connection request for a first socket.

In addition, Applicants respectfully assert that even the Examiner’s assertion as to what is inherent in Beliveau is correct, Beliveau and the Examiner’s statement of inherency still do not teach and enable each and every element of responsive to receiving an incoming connection request for a first socket from said list of sockets that is full, redirecting said connection request to a second socket in said list of sockets that is not full. In particular, Beliveau describes finding the valid application servers for a VIP from the server socket list and selecting one of the valid application servers, where the selection may be enhanced to consider processor loads, delays or other factors.

Beliveau, col. 8, lines 7-18. Even if considering processor loads and delays or other factors could read on detecting that one socket is full and selecting another server, Beliveau still only describes receiving a packet for a VIP and selecting which server of multiple enabled to handled the VIP to connect with for the request; Beliveau, even in view of the Examiner’s statement of what is inherent does not teach or suggest redirecting an incoming connection request for a first socket to another socket.

In addition, Applicants respectfully assert that there is not proper support for the statement that “if a socket is full, inherently, since the system assigns services based upon processor load or delays, the system will assign the service to a second socket” and therefore Beliveau does not teach or enable each and every element of claims 1 and 7. Applicants note that “to serve as an anticipation when the reference is silent about the asserted inherent characteristic, such gap in the reference may be filled with recourse to extrinsic evidence. Such evidence must make clear that the missing descriptive matter is necessarily present in the thing described in the reference, and that

it would be so recognized by persons of ordinary skill.” *Continental Can Co. USA v. Monsanto Co.*, 948 F.2d 1264, 1268, 20 USPQ2d 1746, 1749 (Fed. Cir. 1991). The Examiner has not presented any extrinsic evidence as to how from a system which selects a server to handle non-socket addressed packet through load balancing, it is inherent that such load balancing could be performed to handle an incoming connection request at the socket level. Applicants submit that the gap between Beliveau and what is stated by the Examiner is one that should require extrinsic evidence to support. In particular, Applicants note that in the specification of the present application, Applicants noted in the description of the related art, paragraph 0006, that if a socket queue is full and the socket receives a new connection request, the socket silently discards the connection request and the embodiment of the invention, paragraph 0044, that a kernel will typically set a size limit for each socket queue and discard requests received at a socket if the socket queue is already full. In addition, the specification of the present application, in paragraph 0046 distinguishes that in the present invention, to solve the problem of a socket discarding requests, the kernel detects when a socket is full and when redirecting a connection request to another socket, the kernel may apply load balancing. Applicants are not reading this portion of the specification into the claims, however, Applicants submit the specification of the present invention as evidence that contradicts the Office’s assertion that based on Beliveau’s description of selecting a server to handle a VIP request at an IP level with enhanced selection of a server based on processor load or delay, that the missing elements of a kernel detecting a socket is full and redirecting an incoming connection request to a different socket is necessarily proven and that it would be so recognized by persons of ordinary skill. In view of the lack of support for the assertion of inherency, and the contradiction of this inherency in Applicants’ own specification by one skilled with the art, Applicants respectfully assert that Beliveau fails to teach and enable each and every element of responsive to receiving an incoming connection request for a first socket from said list of sockets that is full, redirecting said connection request to a second socket in said list of sockets that is not full.

Therefore, because Beliveau does not teach each and every element of responsive to receiving an incoming connection request for a first socket from said list of sockets that is full, redirecting said connection request to a second socket in said list of sockets that is not full, Beliveau does not anticipate claims 1 and 7 and the claims should be allowed.

Claims 3-4, 6, 9-10, and 12

Claims 3-4, 6, 9-10, and 12 are dependent claims of independent claims 1 and 7. Since claims 1 and 7 are not anticipated by Beliveau, Beliveau also does not anticipate dependent claims 3-4, 6, 9-10, and 12 and the claims should be allowed.

In addition, with regard to claims 3 and 9, Applicants note that claims 3 and 9 are amended as illustrated by claim 3 to read:

Claim 3 (Currently Amended): The method according to claim 1 for redirecting connection requests further comprising:

responsive to receiving said socket option call at said operating system kernel, binding all of said sockets in said list of sockets to a same port number and setting a separate flag at a socket layer in each separate socket of said list of sockets to designate each of said separate sockets of said list of sockets as a socket which references at least one other socket designated in said list of sockets.

The specification supports the amendments throughout, and for example, in paragraphs 0039, 0040, and 0041, therefore no new matter is presented through the amendments. In addition, Applicants respectfully assert that Beliveau does not teach each and every element of claims 3 and 9 and therefore Beliveau does not anticipate and the claims and the claims should be allowed.

The Office Action cites Beliveau as describing “in the FTP example any incoming port 21 request is bound to the FTP servers in the socket list” and cites col. 8, lines 5-17 as reading on the example. [Office Action, p. 5]

First, Applicants respectfully submit that the Examiner’s conclusion that an incoming request is a port request is incorrect and Applicants submit that the Examiner’s conclusion that an incoming port request is bound to the FTP servers in the

socket list is incorrect. Beliveau describes that for an incoming packet request for a particular VIP address and socket combination, the list may include multiple servers, such as there being multiple FTP servers running for a VIP address.

Second, Applicants respectfully submit that even if Beliveau described “in the FTP example any incoming port 21 request is bound to the FTP servers in the socket list”, this does not teach or enable a kernel, responsive to receiving a socket option call, for binding all the sockets in the list of sockets passed in the call to the same port number. Therefore, Beliveau does not teach or enable responsive to receiving said socket option call at said operating system kernel, binding all of said sockets in said list of sockets to a same port number.

Third, Applicants respectfully submit that no portion of Beliveau teaches or enables setting a separate flag in each of the separate sockets in the list of sockets within the socket call to designate the sockets as one which references at least one other socket. In particular, Applicants respectfully note that Beliveau merely describes maintaining a server socket list, separate from any sockets, which is shared among multiple Fragmenter/De-fragmenters. As previously note, Beliveau does not teach or enable a socket call that includes a list of sockets. In addition, Beliveau does not teach changing any setting of the sockets themselves, at the socket level, based on the socket call. Therefore, Beliveau does not teach or enable each and every element of setting a separate flag at a socket layer in each separate socket of said list of sockets to designate each of said separate sockets of said list of sockets as a socket which references at least one other socket designated in said list of sockets.

Therefore, because Beliveau does not teach and enable each and every element of claims 3 and 9, Applicants respectfully request withdrawal of the rejection under 35 USC 102(e) and allowance of the claims.

Alleged Obviousness under 35 USC 103(a)

Claims 1-4, 6-10, and 12 are not obvious under Beliveau in view of what was allegedly well known in the art at the time of the invention

Claims 1-4, 6-10, and 12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Beliveau in view of what was allegedly well known in the art at the time of the invention. [Office Action, p. 6]

As noted in the Office Action, under 35 USC §103(a) a patent may not be obtained though the invention is not identically disclosed as described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. In *Graham v. John Deere*, the Supreme Court clarified that “under 103, in considering the obviousness or nonobviousness of the subject matter, the scope and content of the prior art are to be determined; differences between the prior art and the claims at issue are to be ascertained; and the level of ordinary skill in the pertinent art resolved, in addition to evaluating evidence of secondary considerations.” *Graham*, 383 U.S. 1, 148 USPQ 459 (1966).

The Examiner bears the initial burden of supporting any prima facie conclusion of obviousness. See *In re Rinehart*, 531, F.2d 1048, 189, USPQ 143 (CCPA 1976); *KSR International Co. v. Teleflex Inc.*, 82 USPQ2d 1385, 1396 (2007); MPEP 2142. The key to supporting a rejection under 35 USC 103 is the clear articulation of the reasons why the claimed invention would have been obvious; the analysis supporting a rejection under 35 USC 103 should be made explicit. See *KSR International Co.*, 82 USPQ2d at 1396; MPEP 2142 (Rev. 6, Sept. 2007).

Applicants traverse the rejection of claims 1-4, 6-10, and 12. Applicants respectfully assert that the Office Action fails to establish a prima facie case of obviousness because the Office erred in the Graham factual findings and there is no clear articulation of the rationale supporting a conclusion of obviousness. Because the Office Action fails to establish a prima facie case of obviousness, Applicants respectfully request withdrawal of the rejection under 35 USC 103(a) and allowance of the claims.

The Office Action rejects claims 1 and 7 on the same grounds. [Office Action, p. 6]
Claim 1, which is representative of claim 7 in grounds of rejection, currently reads:

Claim 1 (Original): A method for redirecting connection requests at an operating system kernel level comprising:

receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service;

setting up all of said sockets in said list of sockets to reference each other in said operating system kernel; and

responsive to receiving an incoming connection request for a first socket from said list of sockets that is full, redirecting said connection request to a second socket in said list of sockets that is not full, such that said operating system kernel redirects said connection request to said second socket providing said same service as said first socket.

Applicants respectfully assert that the Office has erred in finding a prima facie case of obviousness as to claims 1 and 7 because under a proper Graham analysis, when Beliveau is considered as a whole with the Official Notice, the references, do not teach the elements of claims 1 and 7 and there is no clear statement as to the rationale for one of ordinary skill in the art finding claims 1 and 7 as a whole obvious in view of the differences between Beliveau and the Official Notice and claims 1 and 7.

receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service

First, in the Graham inquiry, as to the scope and contents of Beliveau and the difference between Beliveau and claims 1 and 7, Applicants respectfully assert that the Office has committed error in the Graham fact finding regarding the contents of Beliveau and therefore there is not adequate support in the record to support a prima facie case of obviousness against claims 1 and 7.

The Office Action cites the same portions of Beliveau as cited with regard to the rejection under 35 USC 102(e) as reading on the claimed element of receiving, from an application setting up a cluster of servers providing a same service, a socket option call

with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service in the rejection under 35 USC 103(a). [Office Action, p. 7] In particular, as previously noted with regard to Applicants' comments regarding the lack of teaching under 35 USC 102(e), the Office Action again cites column 7, line 62-column 8, line 17 of Beliveau as describing "a socket list is maintained which contains server socket information including service related information, e.g. there was the list of all FTP servers in the cluster of servers each having its own socket in the list" and cites this portion of Beliveau as reading on the claimed element of receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service. [Office Action, p. 7]

Applicants respectfully assert that even if Beliveau's description could factually support the Examiner's assertion that it describes "a socket list is maintained which contains server socket information including service related information", a first difference between Beliveau and claims 1 and 7 is that maintaining a socket list that contains server socket information including service related information does not teach a socket option call with a list of sockets or an operating system kernel receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service.

In addition, Applicants respectfully assert that in view of the actual teachings in Beliveau, there are additional differences between Beliveau and the element of receiving, from an application setting up a cluster of servers providing a same service, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service which are not addressed in the record and therefore the Office errs in concluding a prima facie case of obviousness as to claims 1 and 7. In considering Beliveau as whole, Applicants note that column 7, line 64-column 8, line 3 of Beliveau specify:

when a server uses an API to open a server socket (for example, 80) for a VIP address at step 52, the system call determines that it is a socket for a VIP address. At step 53, the Framework then requests one of the Fragmenter/De-fragmenters to update the list of IPC ports with this new server socket. For the same combination of VIP address/server socket, there may be many IPC ports.

Column 8, lines 23-25 of Beliveau describe the situation where a server uses an API to open a server socket, in that “when an application server such as HTTP-2 needs to establish a socket between itself and a remote client, it first opens a client socket at step 61.” Column 8, lines 4-15 of Beliveau describe:

Therefore, at 54, a server socket list is distributed and shared between all Fragmenter/De-fragmenters. When a packet comes from any source IP address and reaches a Fragmenter/De-fragmenter, the process extracts the destination VIP address and destination socket (for example, 80) from the packet. The Fragmenter/De-fragmenter then finds, through the server socket list, a valid application server. If a plurality of servers can serve this VIP address and server socket combination, the Fragmenter/De-fragmenter selects one of them. For example, if there are six different processors with FTP servers running on them for this VIP address, then the Fragmenter/De-fragmenter selects one of them.”

Applicants respectfully submit that Beliveau’s description of a server using an API to open a server socket for a VIP address, does not however teach a socket option call with a list of sockets. In addition, Beliveau’s description of a socket call through an API triggering a Framework to request a Fragmenter process to add a newly opened socket for a VIP address to an existing list of IPC ports and VIP address/server socket combinations in a server socket list does not teach a socket option call with a list of sockets. Therefore, a difference between Beliveau and claims 1 and 7 is that neither Beliveau’s open socket API call nor Beliveau’s server socket list teach the claimed a socket option call with a list of sockets.

In another difference between Beliveau and claims 1 and 7, when Beliveau is considered as a whole, the list of VIP address/server sockets combinations is not compiled responsive to a single socket option call. Beliveau’s description of maintaining a list of VIP address/server socket combinations which may include multiple servers that

can serve a same VIP address and server socket combination, still lacks the teaching of a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service.

Applicants respectfully submit that because the Office Action fails to assert any rationale as to why, in view of the differences between Beliveau and claims 1 and 7 that claims 1 and 7 would still be obvious to one of ordinary skill in the art at the time of invention, the Office Action fails to establish a prima facie case of obviousness as to claims 1 and 7.

responsive to receiving an incoming connection request for a first socket, redirecting said connection request to a second socket in said list of sockets

Second, in the Graham inquiry, as to the scope and contents of Beliveau and the Official Notice and the difference between Beliveau and claims 1 and 7, Applicants respectfully assert that the Office has committed error in finding facts regarding the scope and content of Beliveau and the Official notice and the differences between Beliveau and the Official Notice and claims 1 and 7 and therefore there is not adequate support in the record to support a prima facie case of obviousness against claims 1 and 7.

In considering the scope and content of Beliveau with regard to the element of responsive to receiving an incoming connection request for a first socket, redirecting said connection request to a second socket in said list of sockets, the Office Action cites column 8, lines 13-17 as describing “since the system assigns services based upon processor load the system may assign the service to a second socket.” [Office Action, p. 7] The Office Action further states:

“Bewiseau may not explicitly or implicitly disclose the redirection of the incoming connection request occurs if the first socket is full and a second socket is not. Even, if this is the case, it would have been obvious to one of ordinary skill in the art that, given Beliveau’s teaching that redirection is assigned based upon round robin selection or may be enhanced to consider processor load, delays or other factors (column 8, lines 13-17), that such a method of load distribution could and should be used.

Therefore, Official Notice (MPEP 2144.01) is taken that it would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Beliveau with a well known practice in the art (i.e. if a resource is busy, and there is a list of resources that can accomplish the same task, utilizing that list to find one that can do it and is not busy) in order to more effectively balance the load in Beliveau's distributed computing system." [Office Action, pp. 7-8]

First, in considering the scope and content of Beliveau, in a Graham inquiry, Applicants respectfully submit that the Office errs by interpreting Beliveau's incoming VIP addressed packet as an equivalent to the claimed incoming connection request to a particular socket. Beliveau describes receiving a packet at a Fragmenter specified to receive packets for the source IP address within the packet, where the Fragmenter extracts a VIP address and socket, finds one or more valid applications servers for the VIP address in the socket server list, and if there is more than one valid application server, selects one of the servers. *Beliveau*, col. 7, lines 10-25, col. 8, lines 5-18. As noted in the Office Action, Beliveau describes that in selecting one of the servers, "the selection may be based on a round-robin selection or may be enhanced to consider processor load, delays, or other factors." Beliveau clearly describes that each incoming request is a packet specified for one of one or more virtual IPs supported within the Framework, and not an incoming connection request for any particular socket. *Beliveau*, col. 1, lines 38-39, col. 4, line 66-col. 5, line 7, col. 7, lines 10-25, col. 8, lines 5-18. Clearly, Beliveau's incoming VIP addressed packets are not an incoming connection request for a particular socket.

Applicants respectfully submit that because the Office Action fails to assert any rationale as to why, in view of the differences between Beliveau and claims 1 and 7 that claims 1 and 7 would still be obvious to one of ordinary skill in the art at the time of invention, the Office Action fails to establish a *prima facie* case of obviousness as to claims 1 and 7.

Second, in considering the scope and content of Beliveau and the Official Notice, in a Graham inquiry, Applicants respectfully assert that the Office errs by concluding that if one reference describes selecting from among multiple servers available for a VIP

address and that the selection may include processor load balancing and Official notice describes that if a resource is busy and there is a list of backup resources, to select a different resource, then one with ordinary skill in the art of operating system kernel design would have combined the references to teach redirecting an incoming connection request from a socket that is full to another socket that is not full, at the socket level. Applicants previously noted that within the specification of the present application, noted in the description of the related art, paragraph 0006, that if a socket queue is full and the socket receives a new connection request, the socket silently discards the connection request and the embodiment of the invention, paragraph 0044, that a kernel will typically set a size limit for each socket queue and discard requests received at a socket if the socket queue is already full. In addition, the specification of the present application, in paragraph 0046 distinguishes that in the present invention, to solve the problem of a socket discarding requests, the kernel detects when a socket is full and when redirecting a connection request to another socket, the kernel may select the redirected socket by load balancing. Even if the specification did not include these teachings, there is nothing in Beliveau or the Official Notice, or the combined references, which indicates any method or process to detect or receive an indicator that a socket is full, at a socket level. Thus, Applicants respectfully assert that even if one with skill in the art at the time of the invention were to combine Beliveau's method for selecting a particular server to handle a VIP request and there were further implemented the Official Notice that if there is a list of backups and a resource is busy, the Office Action still fails to articulate in the rationale why one of ordinary skill in the art would have recognized, in view of the lack of teaching regarding detecting whether a socket is full, that the results of redirecting an incoming connection request if a first socket is full to a second socket were predictable.

In conclusion, Applicants submit that there is no statement of a Graham finding with regard to claims 1 and 7 which articulates a complete set of elements as required to support a prima facie case of obviousness. Because prima facie obviousness is not

establish as to claims 1 and 7, Applicants respectfully request withdrawal of the rejection under 35 USC 103(a) and allowance of the claims.

Claims 2-4, 6, 8-10, and 12

Applicants respectfully assert that because claims 1 and 7 are nonobvious under 35 USC 103(a), claims 2-4, 6, 8-10, and 12 which depend on claims 1 and 7, are also nonobvious and should be allowed. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988).

Claims 5, 11, 13, and 15-16 are not obvious under Beliveau in view of Hopen

Claims 5, 11, 13, and 15-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Beliveau in view of Hopen et al. (US Publication 2005/0132030)(herein referred to as Hopen). [Office Action, p. 11]

As to claims 5 and 11, Applicants respectfully assert that because claims 1 and 7 are nonobvious under 35 USC 103(a), claims 5 and 11 which depend on claims 1 and 7, are also nonobvious and should be allowed. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988).

As to claim 13, Applicants respectfully assert that the Office Action fails to establish a prima facie case of obviousness as to the claims for at least the same reasons that the Office Action fails to establish a prima facie case of obviousness as to claims 1 and 7. In particular, claim 13 reads:

Claim 13 (Currently Amended): A computer program product, residing in a volatile or non-volatile computer readable medium, for redirecting connection requests at an operating system kernel level comprising:

means for enabling receipt, from an application server setting up a master-slave configuration, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets provide a same service;

means for controlling set-up of all of said sockets in said list of sockets to reference each other in said operating system kernel; and

means, responsive to receiving an incoming connection request for a first socket from said list of sockets that is full, for enabling redirection of

said connection request to a second socket in said list of sockets that is not full.

The Office Action cites the same portions of Beliveau as reading on the claimed elements of claim 13 as were cited in the rejection of claims 1 and 7 under 35 USC 102(e), other than with respect to the limitation of an application server setting up a master-slave configuration. [Office Action, pp. 11, 12] The Office Action states that Beliveau does not explicitly disclose setting up a master-slave configuration in the distributed computer system, but states Hopen discloses a computing system that utilizes a master slave configuration. [Office Action, p. 12]

In particular, as to the lack of prima facie case of obviousness, Applicants assert that in the Graham inquiry, as to the scope and contents of Beliveau and the difference between Beliveau and claim 13, Applicants respectfully assert that the Office has committed error in the Graham fact finding regarding the contents of Beliveau and therefore there is not adequate support in the record to support a prima facie case of obviousness against claim 13.

The Office Action again cites column 7, line 62-column 8, line 17 of Beliveau as describing “a socket list is maintained which contains server socket information including service related information, e.g. there was the list of all FTP servers in the cluster of servers each having its own socket in the list” and cites this portion of Beliveau as reading on the claimed element of means for enabling receipt, from an application server, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service. [Office Action, p. 11]

Applicants respectfully assert that even if Beliveau’s description could factually support the Examiner’s assertion that is describes “a socket list is maintained which contains server socket information including service related information”, a first difference between Beliveau and claim 13 is that maintaining a socket list that contains server socket information including service related information does not teach a socket option call with a list of sockets or an operating system kernel means for receiving, from

an application server, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service.

In addition, Applicants respectfully assert that in view of the actual teachings in Beliveau, there are additional differences between Beliveau and the element of means for receiving, from an application server, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service which are not addressed in the record and therefore the Office errs in concluding a prima facie case of obviousness as to claims 1 and 7. In considering Beliveau as whole, Applicants note that column 7, line 64-column 8, line 3 of Beliveau specify:

when a server uses an API to open a server socket (for example, 80) for a VIP address at step 52, the system call determines that it is a socket for a VIP address. At step 53, the Framework then requests one of the Fragmenter/De-fragmenters to update the list of IPC ports with this new server socket. For the same combination of VIP address/server socket, there may be many IPC ports.

Column 8, lines 23-25 of Beliveau describe the situation where a server uses an API to open a server socket, in that “when an application server such as HTTP-2 needs to establish a socket between itself and a remote client, it first opens a client socket at step 61.” Column 8, lines 4-15 of Beliveau describe:

Therefore, at 54, a server socket list is distributed and shared between all Fragmenter/De-fragmenters. When a packet comes from any source IP address and reaches a Fragmenter/De-fragmenter, the process extracts the destination VIP address and destination socket (for example, 80) from the packet. The Fragmenter/De-fragmenter then finds, through the server socket list, a valid application server. If a plurality of servers can serve this VIP address and server socket combination, the Fragmenter/De-fragmenter selects one of them. For example, if there are six different processors with FTP servers running on them for this VIP address, then the Fragmenter/De-fragmenter selects one of them.”

Applicants respectfully submit that Beliveau’s description of a server using an API to open a server socket for a VIP address, does not however teach a socket option

call with a list of sockets. In addition, Beliveau's description of a socket call through an API triggering a Framework to request a Fragmenter process to add a newly opened socket for a VIP address to an existing list of IPC ports and VIP address/server socket combinations in a server socket list does not teach a socket option call with a list of sockets. Therefore, a difference between Beliveau and claim 13 is that neither Beliveau's open socket API call nor Beliveau's server socket list teach the claimed a socket option call with a list of sockets.

In another difference between Beliveau and claim 13, when Beliveau is considered as a whole, the list of VIP address/server sockets combinations is not compiled responsive to a single socket option call. Beliveau's description of maintaining a list of VIP address/server socket combinations which may include multiple servers that can serve a same VIP address and server socket combination, still lacks the teaching of a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service.

Applicants respectfully submit that because the Office Action fails to assert any rationale as to why, in view of the differences between Beliveau and claim 13 that claim 13 would still be obvious to one of ordinary skill in the art at the time of invention, the Office Action fails to establish a prima facie case of obviousness as to claim 13.

As to claim 16, Applicants respectfully assert that because claim 13 is nonobvious under 35 USC 103(a), claim 16 which depends on claim 13, is also nonobvious and should be allowed. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988).

Claims 13-16 are not obvious under Beliveau in view of Hopen and further in view of what was allegedly well known in the art at the time of the invention

Claims 13-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Beliveau in view of Hopen and further in view of what was allegedly well known in the art at the time of the invention. [Office Action, p. 14] Applicants respectfully assert that for at least the same reasons that a prima facie case of obviousness is not established as to claims 1 and 7 under 35 USC 103(a) under Beliveau and further in view of what

was allegedly well known in the art at the time of the invention, a prima facie case of obviousness is also not established as to claim 13. As to claims 14-16, Applicants respectfully assert that because claim 13 is nonobvious under 35 USC 103(a), claims 14-16 which depend on claim 13, are also nonobvious and should be allowed. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988).

In particular, the Office Action rejects claim 13 on the same grounds found in the rejections of claims 1 and 7 under 35 USC 103(a) under Beliveau and further in view of what was allegedly well known in the art, other than as to the element of setting up a master-slave configuration in the distributed computing system. [Office Action, p. 14]

Applicants respectfully assert that the Office has erred in finding a prima facie case of obviousness as to claim 13 because under a proper Graham analysis, when Beliveau is considered as a whole with the Official Notice, the references, do not teach the elements of claim 13 and there is no clear statement as to the rationale for one of ordinary skill in the art finding claim 13 as a whole obvious in view of the differences between Beliveau and the Official Notice and claim 13.

First, for the same reasons that the Office Actions fails to establish a prima facie case of obviousness as to Beliveau in view of Hopen alone with regards to claim 13 as a whole, and in particular with regards to means for enabling receipt, from an application server, a socket option call with a list of sockets for informing an operating system kernel that all of the sockets in said list of sockets will provide said same service, the Office Action also fails to establish a prima facie case of obviousness with regard to claim 13 under Beliveau in view of Hopen and further in view of what was allegedly well known in the art at the time of the invention.

Second, in the Graham inquiry, as to the scope and contents of Beliveau, Hopen, and the Official Notice and the difference between Beliveau, Hopen, the Official Notice, and claim 13, Applicants respectfully assert that the Office has committed error in finding facts regarding the scope and content of Beliveau and the Official notice and the differences between Beliveau and the Official Notice and claim 13 and therefore there is

not adequate support in the record to support a prima facie case of obviousness against claim 13.

In considering the scope and content of Beliveau with regard to the element of means, responsive to receiving an incoming connection request for a first socket from said list of sockets that is full, for enabling redirection of said connection request to a second socket in said list of sockets that is not full, the Office Action cites column 8, lines 13-17 as describing “since the system assigns services based upon processor load the system may assign the service to a second socket.” [Office Action, p. 14] The Office Action further states:

“Bewiseau and Hopen may not explicitly or implicitly disclose that the redirection of the incoming connection request occurs if the first socket is full and a second socket is not. Even, if this is the case, it would have been obvious to one of ordinary skill in the art that, given Beliveau’s teaching that redirection is assigned based upon round robin selection or may be enhanced to consider processor load, delays or other factors (column 8, lines 13-17), that such a method of load distribution could and should be used. Therefore, Official Notice (MPEP 2144.01) is taken that it would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Beliveau and Hopen with a well known practice in the art (i.e. if a resource is busy, and there is a list of resources that can accomplish the same task, utilizing that list to find one that can do it and is not busy) in order to more effectively balance the load in Beliveau’s distributed computing system.” [Office Action, pp. 15-16]

First, in considering the scope and content of Beliveau, in a Graham inquiry, Applicants respectfully submit that the Office errs by interpreting Beliveau’s incoming VIP addressed packet as an equivalent to the claimed incoming connection request to a particular socket. Beliveau describes receiving a packet at a Fragmenter specified to receive packets for the source IP address within the packet, where the Fragmenter extracts a VIP address and socket, finds one or more valid applications servers for the VIP address in the socket server list, and if there is more than one valid application server, selects one of the servers. *Bewiseau*, col. 7, lines 10-25, col. 8, lines 5-18. As noted in the Office Action, Beliveau describes that in selecting one of the servers, “the selection may be based on a round-robin selection or may be enhanced to consider

processor load, delays, or other factors.” Beliveau clearly describes that each incoming request is a packet specified for one of one or more virtual IPs supported within the Framework, and not an incoming connection request for any particular socket.

Believeau, col. 1, lines 38-39, col. 4, line 66-col. 5, line 7, col. 7, lines 10-25, col. 8, lines 5-18. Clearly, Beliveau’s incoming VIP addressed packets are not an incoming connection request for a particular socket.

Applicants respectfully submit that because the Office Action fails to assert any rationale as to why, in view of the differences between Beliveau and claim 13 that claim 13 would still be obvious to one of ordinary skill in the art at the time of invention, the Office Action fails to establish a *prima facie* case of obviousness as to claim 13.

Second, in considering the scope and content of Beliveau and the Official Notice, in a Graham inquiry, Applicants respectfully assert that the Office errs by concluding that if one reference describes selecting from among multiple servers available for a VIP address and that the selection may include processor load balancing and Official notice describes that if a resource is busy and there is a list of backup resources, to select a different resource, then one with ordinary skill in the art of operating system kernel design would have combined the references to teach redirecting an incoming connection request from a socket that is full to another socket that is not full, at the socket level. Applicants previously noted that within the specification of the present application, noted in the description of the related art, paragraph 0006, that if a socket queue is full and the socket receives a new connection request, the socket silently discards the connection request and the embodiment of the invention, paragraph 0044, that a kernel will typically set a size limit for each socket queue and discard requests received at a socket if the socket queue is already full. In addition, the specification of the present application, in paragraph 0046 distinguishes that in the present invention, to solve the problem of a socket discarding requests, the kernel detects when a socket is full and when redirecting a connection request to another socket, the kernel may select the redirected socket by load balancing. Even if the specification did not include these teachings, there is nothing in Beliveau or the Official Notice, or the combined

references, which indicates any method or process which controls detecting or indicating that a socket is full, at a socket level. Thus, Applicants respectfully assert that even if one with skill in the art at the time of the invention were to combine Beliveau's method for selecting a particular server to handle a VIP request and there were further implemented the Official Notice that if there is a list of backups and a resource is busy, the Office Action still fails to articulate in the rationale why one of ordinary skill in the art would have recognized, in view of the lack of teaching regarding detecting whether a socket is full, that the results of redirecting an incoming connection request if a first socket is full to a second socket were predictable.

In conclusion, Applicants submit that there is no statement of a Graham finding with regard to claim 13 which articulates a complete set of elements as required to support a prima facie case of obviousness. Because prima facie obviousness is not establish as to claim 13, Applicants respectfully request withdrawal of the rejection under 35 USC 103(a) and allowance of the claims.

Conclusion

Applicants note the citation of pertinent prior art cited by the Examiner.

In view of the foregoing, withdrawal of the rejections and the allowance of the current pending claims is respectfully requested. If the Examiner feels that the pending claims could be allowed with minor changes, the Examiner is invited to telephone the undersigned to discuss an Examiner's Amendment.

No extension of time is believed to be necessary. If, however, an extension of time is required, the undersigned hereby authorizes the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0447.

Respectfully submitted,

By Amy J. Pattillo, Reg. No. 46,983/

AMY J. PATTILLO
Registration No. 46,983
P.O. BOX 161327
AUSTIN, TEXAS 78716
ATTORNEY FOR APPLICANTS
Telephone: 512-402-9820
Facsimile: 512-306-0417